CLAIMS

We claim:

1. A system for an integrated circuit card interface device, said interface device operable in one or more modes of operation, comprising:

an application memory;

an application engine for managing one or more applications in said application memory;

an input/output module;

a host interface;

one or more integrated circuit card interfaces;

a display unit; and

an input unit;

wherein said application engine interacts with said input/output module appropriate to said mode of operation to interface with an integrated circuit card operably connected to said interface device.

2. A system as in claim 1, wherein said application memory further comprises a read-only memory.

3. A system as in claim 1, wherein said application memory further comprises an electrically erasable programmable read-only memory.

4. A system as in claim 1, wherein said application engine further comprises a microcontroller.

5. A system as in claim 4, wherein said microcontroller further comprises said application memory.

6. A system as in claim 1, wherein said input/output module comprises a microcontroller.

7. A system as in claim 1, wherein said application engine further comprises a custom circuit.

8. A system as in claim 7, wherein said custom circuit further comprises said application memory.

9. A system as in claim 1, wherein said input/output module further comprises a custom circuit.

10. A method for an integrated circuit card interface device to communicate with an integrated circuit card in connected mode, said interface device operable in one or more modes of operation, comprising the steps of:

receiving a command in said interface device from a host device, said host device connected to said interface device;

performing first processing in response to said command in an I/O module in said interface device;

in response to input from said I/O module, optionally performing one or more additional processing steps in response to said command; and

completing processing of said command.

11. A method as in claim 10, wherein said I/O module performs said one or more additional processing steps.

12. A method as in claim 10, wherein said application engine performs said one or more additional processing steps.

13. A method as in claim 10, wherein said one or more additional processing steps are performed in said I/O module and in said application engine.

14. A method for an integrated circuit card interface device to communicate with an integrated circuit card in standalone mode, said interface device containing multiple modes of operation, comprising the steps of:

receiving an interrupt in said interface device in response to a signal originated by a user of said interface device;

initializing resources in said interface device to respond to said interrupt;

processing said interrupt; and

completing processing in response to said interrupt.

15. A method as in claim 14, wherein said interrupt is generated in response to said user pressing a key.

16. A method as in claim 14, wherein said processing further comprises the steps of:

reading data from a real-time clock; and

displaying said data on a display unit.

17. A method as in claim 16, wherein said data further comprises a time value.

18. A method as in claim 16, wherein said data further comprises a date value.

19. A method as in claim 14, wherein said processing further comprises the steps of:

providing a calculator function;

accepting input from a user;

displaying said input to said user on said display unit;

calculating results of said input; and

displaying the result of the calculations performed on said input on a display unit.

20. A method for an integrated circuit card interface device to communicate with an integrated circuit card in standalone mode, said interface device operable in one or more modes of operation, comprising the steps of:

executing an interface device application program in an application engine within said interface device;

receiving a command from said application engine in an I/O module within said interface device;

performing first processing in response to said command in an I/O module in said interface device;

in response to input from said I/O module, optionally performing one or more additional processing steps in response to said command; and

completing execution of said interface device application program.

21. A method as in claim 20, wherein said I/O module performs said one or more additional processing steps.

22. A method as in claim 20, wherein said application engine performs said one or more additional processing steps.

23. A method as in claim 20, wherein said one or more additional processing steps are performed in said I/O module and in said application engine.

24. A method for updating a program within an integrated circuit card interface device in connected mode, said interface device containing multiple modes of operation, comprising the steps of:

receiving a command from a host device in an I/O module within said interface device, said command instructing said interface device to begin said updating process;

establishing a communications interface between said I/O module and an application engine within said interface device;

determining whether said program can be stored in one or more memory subsets contained within said interface device;

downloading said program to one or more of said memory subsets;

updating reference information applicable to said interface device; and

confirming successful download of said program to one or more of said memory subsets in said interface device.

25. A method as in claim 24, wherein said memory subsets comprise pages within nonvolatile memory.

26. A method as in claim 25, wherein said nonvolatile memory comprises EEPROM.

27. A method as in claim 24, wherein said downloading step further comprises the steps of;

retrieving said program from local storage within said host device; and

transferring said program to said interface device.

28. A method as in 27, wherein said local storage further comprises a hard drive.

29. A method as in claim 24, wherein said downloading step further comprises the steps of:

establishing a communications channel with a remote site;

receiving said program in said host device from said remote site;

storing said program in local storage within said host device;

retrieving said program from local storage within said host device; and

transferring said program to said interface device.

30. A method as in claim 24, wherein said confirming step further comprises the step of checking the integrity of said program.

31. A method as in claim 30, wherein said checking step further comprises the steps of:

calculating a first checksum on said program; and

comparing said first checksum to a second checksum included with said program.

32. A method as in claim 30, wherein said checking step further comprises the step of verifying a digital signature on said program.

33. A method as in claim 24, wherein said confirming step further comprises the step of checking the authenticity of said program.

34. A method as in claim 33, wherein said checking step further comprises the step of verifying a digital signature on said program.

35. A system for updating a program within an integrated circuit card interface device in connected mode, said interface device containing multiple modes of operation, comprising:

a host device;

means for establishing a communications channel between said host device and said interface device;

means for receiving a command from a host device in an I/O module within said interface device, said command instructing said interface device to begin said updating process;

a communications interface between said I/O module and an application engine within said interface device;

means for determining whether said program can be stored in one or more memory subsets contained within said interface device;

means for downloading said program to one or more of said memory subsets;

means for updating reference information applicable to said interface device; and

means for confirming the successful download of said executable application to one or more of said memory subsets in said interface device.

36. A system as in claim 35, wherein said application engine further comprises a microcontroller.

37. A system as in claim 36, wherein said microcontroller further comprises a flash microcontroller.

38. A system as in claim 35, wherein said input/output module further comprises a microcontroller.

39. A system as in claim 35, wherein said application engine further comprises a custom circuit.

40. A system as in claim 35, wherein said input/output module comprises a custom circuit.